**LAPORAN PRAKTIKUM 7**

**STRUKTUR DATA**

**A logo of a university

AI-generated content may be incorrect.**

Dosen Pengampu:  
Dr. Wahyudi, S.T, M.Kom.

Disusun Oleh:  
Muhammad Luthfi Kautsar Rizata – 2311532020

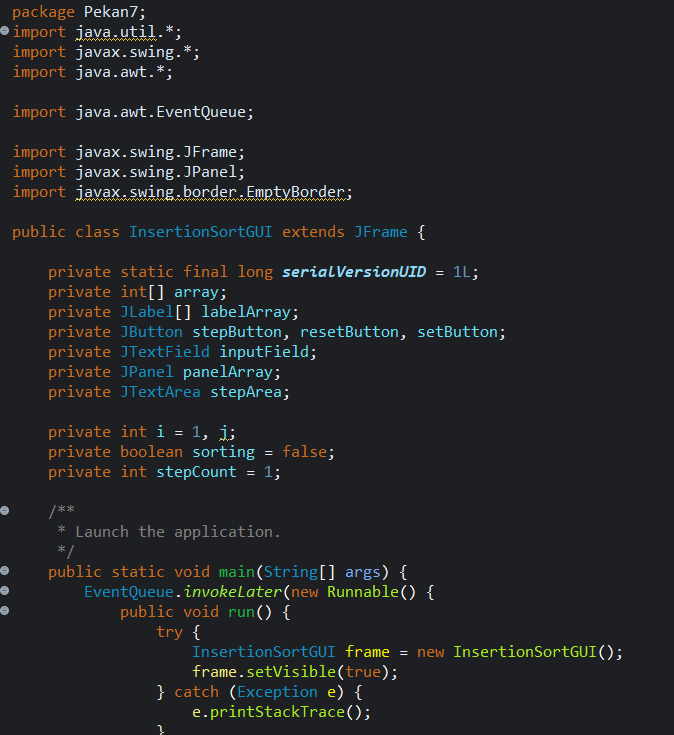
**DEPARTEMEN INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS ANDALAS**

**2025**

**A screen shot of a computer program

AI-generated content may be incorrect.****1. Class InsertionSortGUI**

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A computer screen shot of colorful text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Program InsertionSortGUI merupakan aplikasi berbasis Java Swing yang dirancang untuk memvisualisasikan proses algoritma *Insertion Sort* secara interaktif dan bertahap. Program ini memungkinkan pengguna untuk memasukkan deretan angka yang akan diurutkan melalui sebuah kolom input teks, di mana angka-angka tersebut dipisahkan menggunakan tanda koma. Setelah angka dimasukkan dan tombol “Set Array” ditekan, program akan menampilkan elemen-elemen array tersebut dalam bentuk label yang tersusun horizontal, sehingga pengguna dapat melihat posisi masing-masing angka secara visual. Komponen label ini berada di dalam sebuah panel yang disebut panelArray.

Ketika pengguna menekan tombol “Langkah Selanjutnya”, program akan menjalankan satu langkah dari algoritma *Insertion Sort*—yaitu memilih satu elemen dan memindahkannya ke posisi yang sesuai di subarray yang sudah terurut di sebelah kirinya. Proses ini dilakukan satu per satu mulai dari indeks ke-1 hingga akhir array. Setiap langkah dicatat dan ditampilkan dalam *text area* di sisi kanan jendela aplikasi agar pengguna dapat membaca log proses yang sedang berjalan. Elemen yang dipindahkan akan digeser, dan tampilan array diperbarui untuk mencerminkan perubahan tersebut.

Program ini juga menyediakan tombol “Reset” yang memungkinkan pengguna menghapus semua input dan memulai proses dari awal. Ketika proses sorting selesai, tombol “Langkah Selanjutnya” akan otomatis dinonaktifkan dan akan muncul jendela pop-up yang memberitahu bahwa proses telah selesai. Dengan cara kerja seperti ini, aplikasi ini sangat cocok digunakan untuk pembelajaran algoritma, karena setiap perubahan dapat dilihat secara langsung dan pengguna bisa memahami bagaimana *Insertion Sort* bekerja secara bertahap, bukan sekaligus.

Secara internal, program menggunakan beberapa variabel penting, seperti array untuk menyimpan data angka yang dimasukkan, labelArray untuk menampung label visual angka, serta variabel i dan j sebagai indeks dalam algoritma sorting. Proses sorting hanya dimulai setelah tombol "Set Array" ditekan, dan hanya akan berjalan satu langkah setiap kali tombol “Langkah Selanjutnya” diklik. Selain itu, setiap langkah menampilkan pesan log seperti "Langkah 1: Memasukkan 4" agar pengguna bisa mengikuti proses berpikir dari algoritma. Dengan desain interaktif dan tampilan visual yang sederhana, program ini menjembatani pemahaman konseptual algoritma dengan visualisasi yang konkret.

Pada gambar bagian Desgin menampilkan antarmuka dari program GUI dengan judul "Insertion Sort Langkah per Langkah". Di bagian atas terdapat input field untuk memasukkan deretan angka yang dipisahkan koma, contohnya seperti 2,6,7,8,3. Setelah pengguna menekan tombol "Set Array", deretan angka tersebut divisualisasikan dalam bentuk kotak-kotak horizontal. Setiap elemen array ditampilkan dalam kotak dengan ukuran dan font yang cukup besar agar mudah dibaca.

Di sisi kanan, terdapat area teks yang mencatat log setiap langkah algoritma Insertion Sort. Misalnya, pada langkah pertama ditampilkan "Langkah 1: Memasukkan 6", dan hasil array saat ini. Begitu juga untuk langkah-langkah selanjutnya, seperti memasukkan 7, 8, hingga akhirnya angka 3 dimasukkan ke dalam posisi yang benar. Catatan log ini membantu pengguna memahami bagaimana Insertion Sort bekerja dari satu elemen ke elemen berikutnya, dengan pendekatan yang jelas dan sistematis.

Tombol “Langkah Selanjutnya” di bagian bawah memungkinkan pengguna untuk mengontrol proses sorting secara manual, satu langkah per klik. Hal ini memberi pengguna kesempatan untuk mengamati perubahan array secara bertahap dan memperdalam pemahaman tentang proses penyisipan elemen ke tempat yang tepat. Setelah proses sorting selesai, muncul jendela dialog dengan pesan “Sorting selesai!” sebagai penanda bahwa seluruh elemen sudah diurutkan dengan benar.

A screen shot of a computer program

AI-generated content may be incorrect.A computer screen shot of a program code

AI-generated content may be incorrect.**2. Class SelectionSortGUI**

A screen shot of a computer code

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Program SelectionSortGUI ini merupakan implementasi algoritma *Selection Sort* dalam bentuk visual menggunakan antarmuka grafis berbasis Java Swing. Pengguna dapat memasukkan deretan angka yang dipisahkan dengan koma, lalu melihat proses pengurutan dilakukan langkah demi langkah dengan menekan tombol "Langkah Selanjutnya". Visualisasi array ditampilkan menggunakan label-label yang menunjukkan elemen array, dan langkah-langkah sorting dicatat dalam area teks di sisi kanan jendela.

Secara umum, program ini sudah berjalan dengan baik dan memiliki antarmuka yang cukup informatif. Namun, ada beberapa bagian yang perlu diperhatikan lebih lanjut. Salah satu hal yang cukup krusial adalah inisialisasi variabel j di dalam method setArrayFromInput(). Pada bagian ini terdapat kode j = j + 1;, padahal j belum diberi nilai awal secara eksplisit. Hal ini dapat menyebabkan nilai j tidak konsisten, apalagi jika pengguna beberapa kali mengatur ulang array. Seharusnya, j langsung di-set menjadi i + 1 untuk menjaga urutan langkah algoritma tetap benar.

Selain itu, ketika proses penukaran elemen terjadi, log yang ditampilkan mencetak nilai array setelah pertukaran dilakukan. Hal ini membuat informasi yang ditampilkan menjadi tidak akurat, karena nilai elemen yang tertukar telah berubah saat log dicetak. Akan lebih baik jika nilai sebelum pertukaran disimpan terlebih dahulu dalam variabel sementara, lalu dicetak ke log sebelum proses swap dilakukan.

Visualisasi yang ditampilkan saat ini hanya menyoroti elemen dengan indeks minimum (minIndex), sementara posisi i dan j tidak ditandai. Padahal, dalam konteks pemahaman algoritma, akan sangat membantu jika pengguna dapat melihat elemen yang sedang dibandingkan (yaitu j) dan elemen yang sedang dipertimbangkan sebagai posisi minimum (i). Penambahan warna highlight yang berbeda untuk masing-masing peran ini akan membuat visualisasi jauh lebih mudah dipahami.

Program juga belum menangani input kosong atau tidak valid secara sempurna. Jika pengguna memasukkan input seperti ,, atau , ,, maka program tetap akan mencoba menjalankan sorting, yang bisa menyebabkan error atau hasil yang tidak diharapkan. Perlu ada validasi tambahan untuk memastikan bahwa setiap elemen dalam array yang dimasukkan benar-benar merupakan angka.

Terlepas dari kekurangan tersebut, program ini tetap menunjukkan struktur yang rapi dan penggunaan komponen GUI yang tepat. Desainnya sederhana, intuitif, dan cocok untuk tujuan edukatif. Jika dikembangkan lebih lanjut, program ini bisa diperluas dengan fitur-fitur tambahan seperti animasi otomatis dengan delay antar langkah, mode sorting menurun (descending), atau kemampuan pause dan resume untuk auto-sorting. Dengan peningkatan seperti itu, program ini dapat menjadi alat pembelajaran sorting algoritma yang sangat efektif.

Pada Gambar Design menunjukkan antarmuka dari program GUI dengan judul “Selection Sort Langkah per Langkah”. Program ini dirancang untuk membantu pengguna memahami proses Selection Sort secara visual dan bertahap. Di bagian atas terdapat kolom input untuk memasukkan angka-angka yang dipisahkan koma, misalnya 5,8,6,3,2,9, yang kemudian divisualisasikan dalam bentuk kotak angka yang tersusun rapi secara horizontal. Setelah tombol “Set Array” ditekan, pengguna bisa melihat proses penyortiran satu per satu dengan mengklik tombol “Langkah Selanjutnya”.

Pada area sebelah kanan, terdapat log proses Selection Sort yang mencatat langkah demi langkah. Misalnya, di Langkah 2 tertulis bahwa elemen ke-1 (yaitu angka 8) ditukar dengan angka 3, karena 3 adalah nilai minimum di sisa array. Setelah pertukaran, array berubah menjadi 2, 3, 6, 8, 5, 9. Kemudian di Langkah 3 terjadi lagi pertukaran antara elemen ke-2 (6) dengan 5, sehingga array menjadi 2, 3, 5, 8, 6, 9. Proses ini terus berlanjut hingga seluruh elemen berada pada posisi yang benar.

Menariknya, ketika sudah tidak ditemukan elemen yang perlu ditukar—misalnya saat elemen ke-4 ternyata sudah minimum—program mencatat bahwa tidak ada pertukaran, menunjukkan bahwa algoritma mengenali kondisi optimal di tengah proses. Di akhir penyortiran, muncul jendela dialog bertuliskan “Sorting Selesai!” sebagai penanda bahwa array sudah dalam kondisi terurut sempurna, yaitu 2, 3, 5, 6, 8, 9.